To estimate how long it will take to solve a problem given the running times for various input sizes, we first need to analyze the pattern in the provided data and determine the likely time complexity of the algorithm.

Given data:

- For input size $\mathrm{n} = 1000$, time $\mathrm{T}(n) = 5$ seconds
- For input size $\mathrm{n} = 2000$, time $\mathrm{T}(n) = 20$ seconds
- For input size $\mathrm{n} = 3000$, time $\mathrm{T}(n) = 45$ seconds
- For input size $\mathrm{n} = 4000$, time $\mathrm{T}(n) = 80$ seconds

Let's examine the increase in time as the input size increases. Here are the differences between consecutive times:

- From 1000 to 2000: $20 - 5 = 15$ seconds
- From 2000 to 3000: $45 - 20 = 25$ seconds
- From 3000 to 4000: $80 - 45 = 35$ seconds

We observe that the differences themselves are increasing, which suggests that the time complexity might be more than linear. Let's further look at the rate of increase of these differences:

- Difference between differences (second differences):
  - From 15 to 25: $25 - 15 = 10$
  - From 25 to 35: $35 - 25 = 10$

The second differences are constant, which indicates that the time complexity is likely quadratic. If the second differences are constant, this suggests the function $\mathrm{T}(n)$ follows a quadratic relationship of the form $\mathrm{T}(n) = an^2 + bn + c$.

To find the coefficients a, b, and c, we can set up a system of equations using the given data points. We have:

1. $\mathrm{T}(1000) = 5$
2. $\mathrm{T}(2000) = 20$
3. $\mathrm{T}(3000) = 45$
4. $\mathrm{T}(4000) = 80$

Plugging these into the quadratic form:

1. $\mathrm{a}(1000)^2 + b(1000) + c = 5$
2. $\mathrm{a}(2000)^2 + b(2000) + c = 20$
3. $\mathrm{a}(3000)^2 + b(3000) + c = 45$
4. $\mathrm{a}(4000)^2 + b(4000) + c = 80$

Simplifying each:

1. $1000000a + 1000b + c = 5$
2. $4000000a + 2000b + c = 20$
3. $9000000a + 3000b + c = 45$
4. $16000000a + 4000b + c = 80$

Let's solve this system step-by-step. First, subtract the first equation from the second, the second from the third, and the third from the fourth to eliminate c:

1. $(4000000a + 2000b + c) - (1000000a + 1000b + c) = 20 - 5$
   - $3000000a + 1000b = 15$ (Equation 5)
2. $(9000000a + 3000b + c) - (4000000a + 2000b + c) = 45 - 20$
   - $5000000a + 1000b = 25$ (Equation 6)
3. $(16000000a + 4000b + c) - (9000000a + 3000b + c) = 80 - 45$
   - $7000000a + 1000b = 35$ (Equation 7)

Next, subtract Equation 5 from Equation 6 and Equation 6 from Equation 7:

1. $(5000000a + 1000b) - (3000000a + 1000b) = 25 - 15$

   - $2000000a = 10$
   - $\mathrm{a} = \frac{10}{2000000} = 0.000005$
2. $(7000000a + 1000b) - (5000000a + 1000b) = 35 - 25$

   - $2000000a = 10$
   - $\mathrm{a} = \frac{10}{2000000} = 0.000005$

Both solutions for a match, so we confirm $\mathrm{a} = 0.000005$.

Using Equation 5 to find b:

- $3000000(0.000005) + 1000b = 15$
- $15 + 1000b = 15$
- $1000b = 0$
- $\mathrm{b} = 0$

Now, using one of the original equations to find c:

- $1000000(0.000005) + 1000(0) + c = 5$
- $5 + c = 5$
- $\mathrm{c} = 0$

Thus, the quadratic model is $\mathrm{T}(n) = 0.000005n^2$.

Finally, to estimate $\mathrm{T}(5000)$:

- $\mathrm{T}(5000) = 0.000005(5000)^2 = 0.000005 \times 25000000 = 125$

So, the algorithm will take approximately 125 seconds for an input size of 5000. The algorithm follows a quadratic time complexity.